

Отчет по лабораторной работе №8

Дисциплина: архитектура компьютера

Клименко Алёна Сергеевна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
4.1	Реализация циклов в NASM	8
4.2	Обработка аргументов командной строки	14
4.3	Задание для самостоятельной работы	20
5	Выводы	24
6	Список литературы	25

Список иллюстраций

4.1	Создание каталога	8
4.2	Копирование программы из листинга	9
4.3	Запуск программы	10
4.4	Изменение программы	11
4.5	Запуск измененной программы	12
4.6	Добавление push и pop в цикл программы	13
4.7	Запуск измененной программы	14
4.8	Копирование программы из листинга	15
4.9	Запуск второй программы	16
4.10	Копирование программы из третьего листинга	17
4.11	Запуск третьей программы	18
4.12	Изменение третьей программы	19
4.13	Запуск измененной третьей программы	20
4.14	Написание программы для самостоятельной работы	21
4.15	Запуск программы для самостоятельной работы	23

Список таблиц

1 Цель работы

Приобретение навыков написания программ с использованием циклов и обработкой аргументов командной строки.

2 Задание

1. Реализация циклом в NASM
2. Обработка аргументов командной строки
3. Самостоятельное написание программы по материалам лабораторной работы

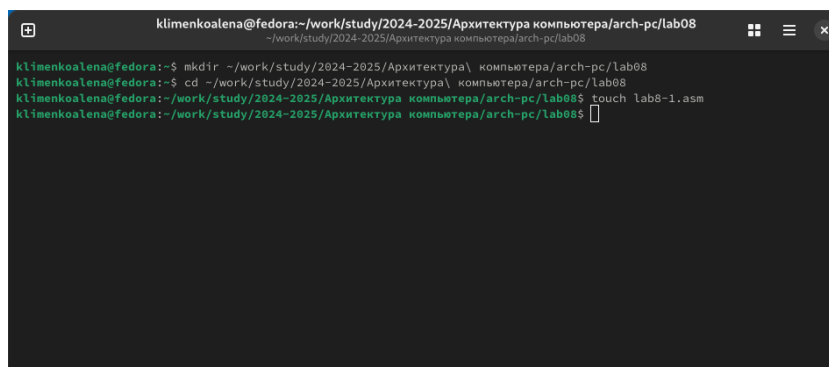
3 Теоретическое введение

Стек — это структура данных, организованная по принципу LIFO («Last In — First Out» или «последним пришёл — первым ушёл»). Стек является частью архитектуры процессора и реализован на аппаратном уровне. Для работы со стеком в процессоре есть специальные регистры (ss, bp, sp) и команды. Основной функцией стека является функция сохранения адресов возврата и передачи аргументов при вызове процедур. Кроме того, в нём выделяется память для локальных переменных и могут временно храниться значения регистров.

4 Выполнение лабораторной работы

4.1 Реализация циклов в NASM

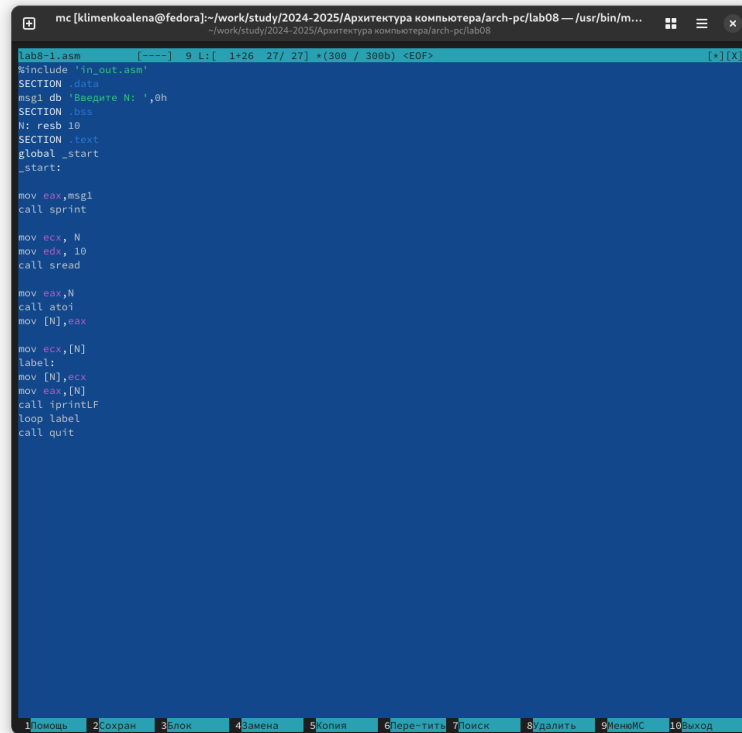
Создаю каталог для программ лабораторной работы №8, перехожу в него и создаю файл lab8-1.asm . (рис. 4.1).



```
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08
klimenkoalena@fedora:~$ mkdir ~/work/study/2024-2025/Архитектура\ компьютера/arch-pc/lab08
klimenkoalena@fedora:~$ cd ~/work/study/2024-2025/Архитектура\ компьютера/arch-pc/lab08
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ touch lab8-1.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$
```

Рис. 4.1: Создание каталога

Копирую в созданный файл программу из листинга. (рис. 4.2).



```
mc [klimenkoalena@fedora]~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08 — /usr/bin/m...
lab8-1.asm [-----] 9 L: [ 1+26 27/ 27] *(300 / 300b) <EOF> [•] [X]
#include "in_out.asm"
SECTION .data
msg1 db "Введите N: ",0h
SECTION .bss
N: resb 10
SECTION .text
global _start
_start:
mov eax,msg1
call sprint

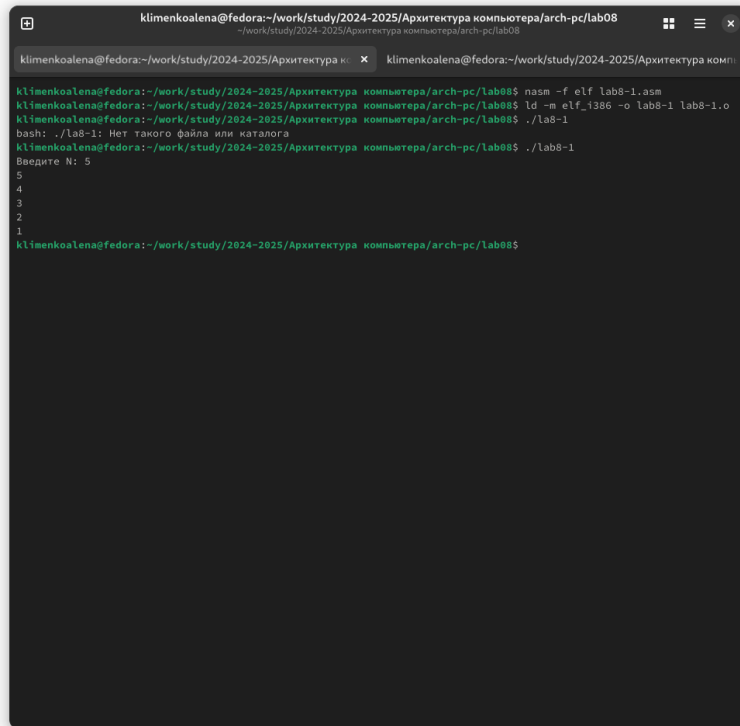
mov ecx,N
mov edx,10
call sread

mov eax,N
call atoi
mov [N],eax

mov ecx,[N]
label:
mov [N],ecx
mov eax,[N]
call iprintf
loop label
call quit
```

Рис. 4.2: Копирование программы из листинга

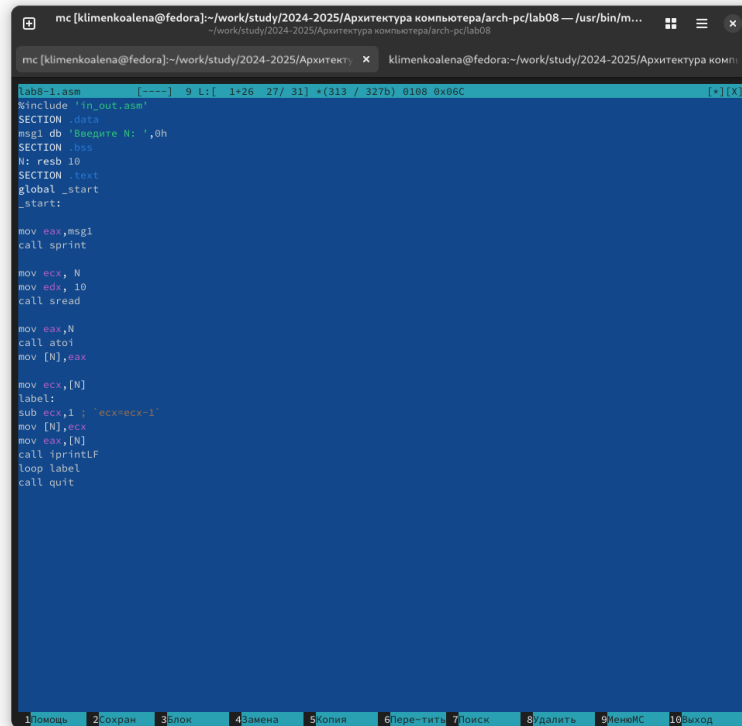
Запускаю программу, она показывает работу циклов в NASM (рис. 4.3).



```
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-1.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-1
bash: ./lab8-1: Нет такого файла или каталога
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-1
Введите N: 5
5
4
3
2
1
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$
```

Рис. 4.3: Запуск программы

Заменяю программу изначальную так, что в теле цикла я изменяю значение регистра есх (рис. 4.4).



```
lab8-1.asm [-----] 9 L: [ 1+26 27/ 31] *(313 / 327b) 0108 0x06C [*] [X]
#include "in_out.asm"
SECTION .data
msg1 db "Введите N: ",0h
SECTION .bss
N: resb 10
SECTION .text
global _start
_start:

mov eax,msg1
call sprint

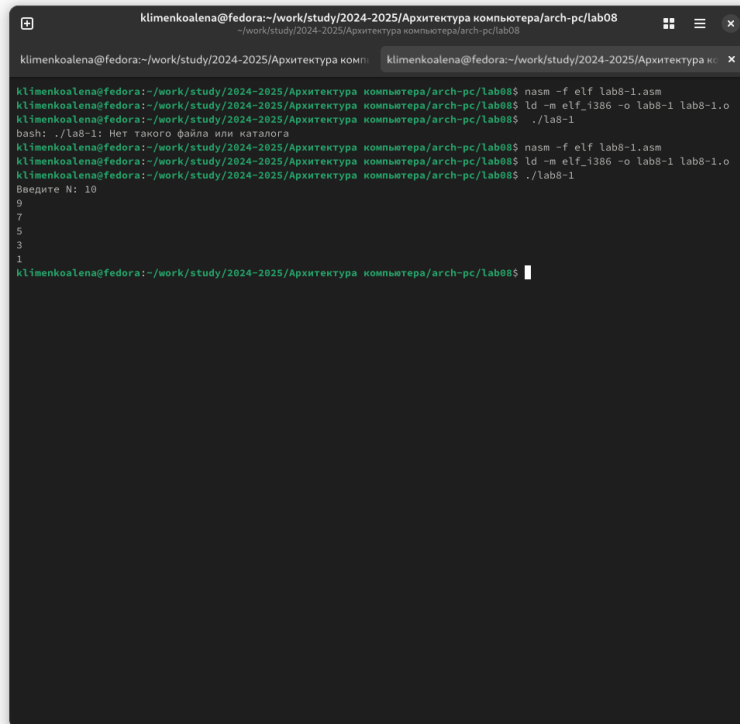
mov ecx, N
mov edx, 10
call sread

mov eax,N
call atoi
mov [N],eax

mov ecx,[N]
label:
sub ecx,1 ; "ecx=ecx-1"
mov [N],ecx
mov eax,[N]
call iprintLF
loop label
call quit
```

Рис. 4.4: Изменение программы

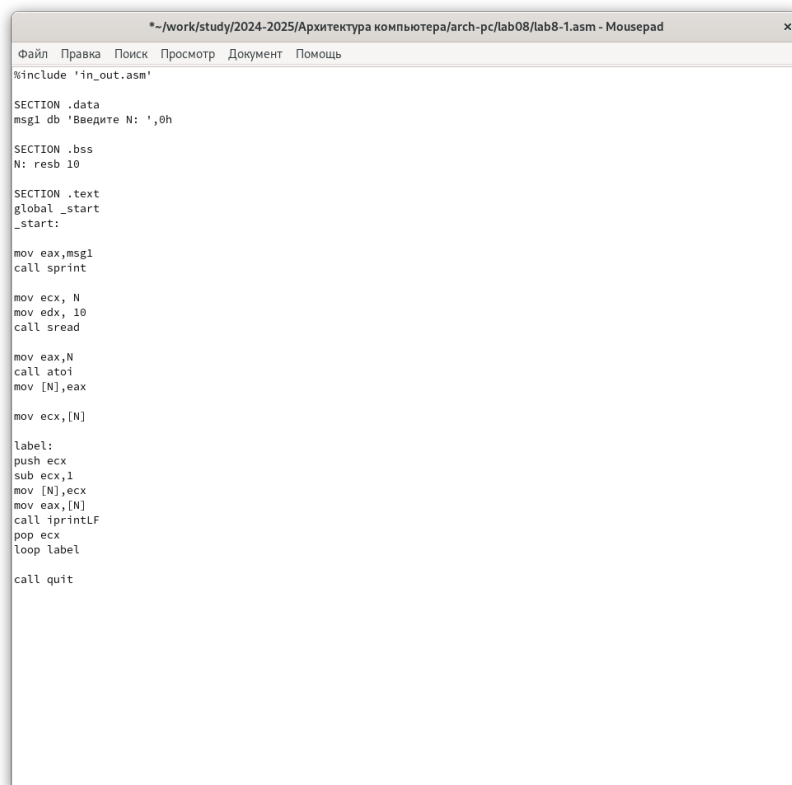
Из-за того, что теперь регистр ecx на каждой итерации уменьшается на 2 значения, количество итераций уменьшается вдвое (рис. 4.5).



```
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура комп: klimenkoalena@fedora:~/work/study/2024-2025/Архитектура к
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-1.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-1
bash: ./lab8-1: Нет такого файла или каталога
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-1.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-1
Введите N: 10
9
7
5
3
1
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$
```

Рис. 4.5: Запуск измененной программы

Добавляю команды push и pop в программу (рис. 4.6).



```
*~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08/lab8-1.asm - Mousepad
Файл  Правка  Поиск  Просмотр  Документ  Помощь

%include 'in_out.asm'

SECTION .data
msg1 db 'Введите N: ',0h

SECTION .bss
N: resb 10

SECTION .text
global _start
_start:

mov eax,msg1
call sprint

mov ecx, N
mov edx, 10
call sread

mov eax,N
call atoi
mov [N],eax

mov ecx,[N]

label:
push ecx
sub ecx,1
mov [N],ecx
mov eax,[N]
call iprintLF
pop ecx
loop label

call quit
```

Рис. 4.6: Добавление push и pop в цикл программы

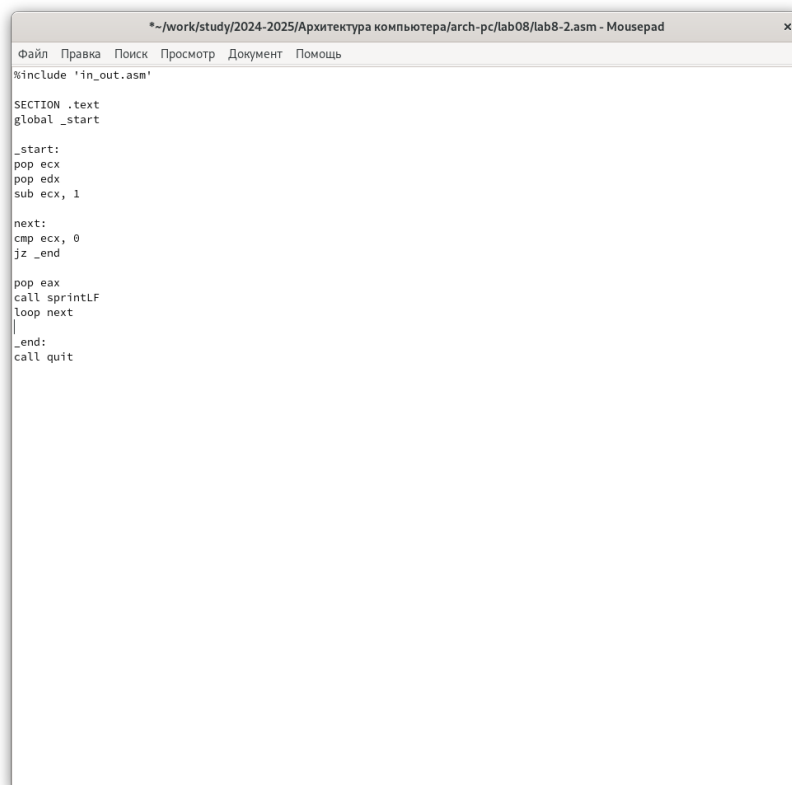
Теперь количество итераций совпадает введенному N, но произошло смещение выводимых чисел на -1 (рис. 4.7).

```
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура комп: klimenkoalena@fedora:~/work/study/2024-2025/Архитектура к
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-1.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-1
bash: ./lab8-1: Нет такого файла или каталога
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-1.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-1
Введите N: 10
9
8
7
6
5
4
3
2
1
0
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ mousepad lab8-1.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-1.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_i386 -o lab8-1 lab8-1.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-1
Введите N: 10
9
8
7
6
5
4
3
2
1
0
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$
```

Рис. 4.7: Запуск измененной программы

4.2 Обработка аргументов командной строки

Создаю новый файл для программы и копирую в него код из следующего листинга (рис. 4.8).

A screenshot of a text editor window titled '*~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08/lab8-2.asm - Mousepad'. The window has a menu bar with 'Файл', 'Правка', 'Поиск', 'Просмотр', 'Документ', and 'Помощь'. The main text area contains the following assembly code:

```
%include 'in_out.asm'

SECTION .text
global _start

_start:
pop ecx
pop edx
sub ecx, 1

next:
cmp ecx, 0
jz _end

pop eax
call sprintf
loop next
|
_end:
call quit
```

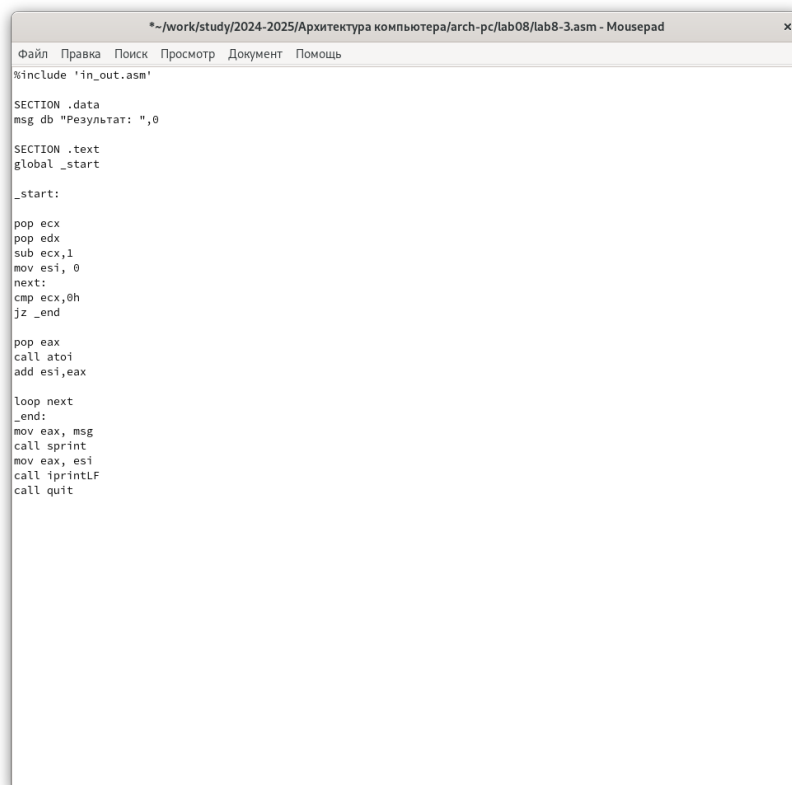
Рис. 4.8: Копирование программы из листинга

Компилирую программу и запускаю, указав аргументы. Программой было об-
ратено то же количество аргументов, что и было введено (рис. 4.9).

```
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ touch lab8-2.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ mousepad lab8-2.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf_1386 -o lab8-2 lab8-2.o
nasm: fatal: unrecognised output format 'elf_1386' - use -hf for a list
Type nasm -h for help.
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f wlf lab8-2.asm
nasm: fatal: unrecognised output format 'wlf' - use -hf for a list
Type nasm -h for help.
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-2.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_1386 -o lab8-2 lab-2.o
ld: невозможно найти lab-2.o: Нет такого файла или каталога
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_1386 -o lab8-2 lab8-2.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-2 arg1 arg 2 'arg 3'
arg1
arg
2
arg 3
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$
```

Рис. 4.9: Запуск второй программы

Создаю новый файл для программы и копирую в него код из третьего листинга (рис. 4.10).

A screenshot of a text editor window titled "*~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08/lab8-3.asm - Mousepad". The window has a menu bar with "Файл", "Правка", "Поиск", "Просмотр", "Документ", and "Помощь". The code content is as follows:

```
%include 'in_out.asm'

SECTION .data
msg db "Результат: ",0

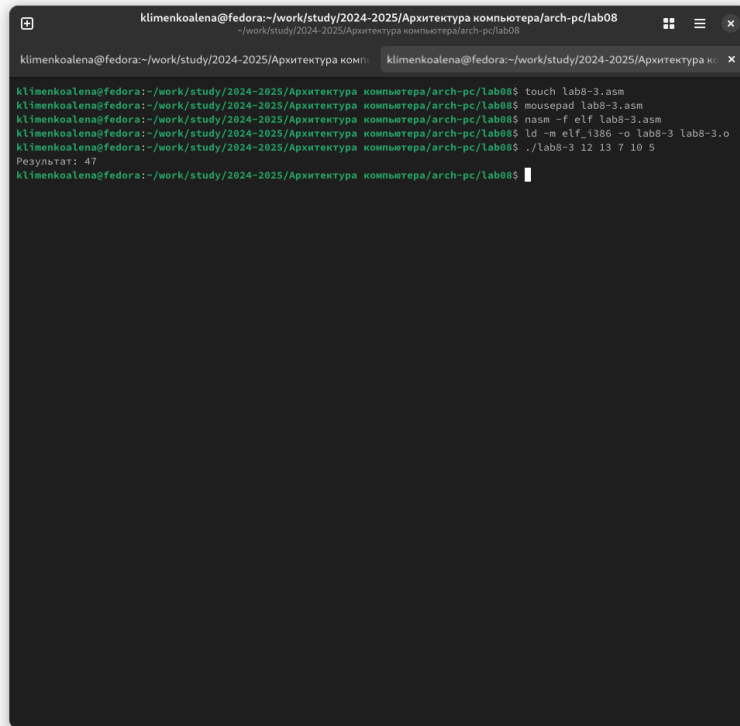
SECTION .text
global _start

_start:
    pop ecx
    pop edx
    sub ecx,1
    mov esi, 0
next:
    cmp ecx,0h
    jz _end
    pop eax
    call atoi
    add esi,eax

    loop next
_end:
    mov eax, msg
    call sprint
    mov eax, esi
    call iprintLF
    call quit
```

Рис. 4.10: Копирование программы из третьего листинга

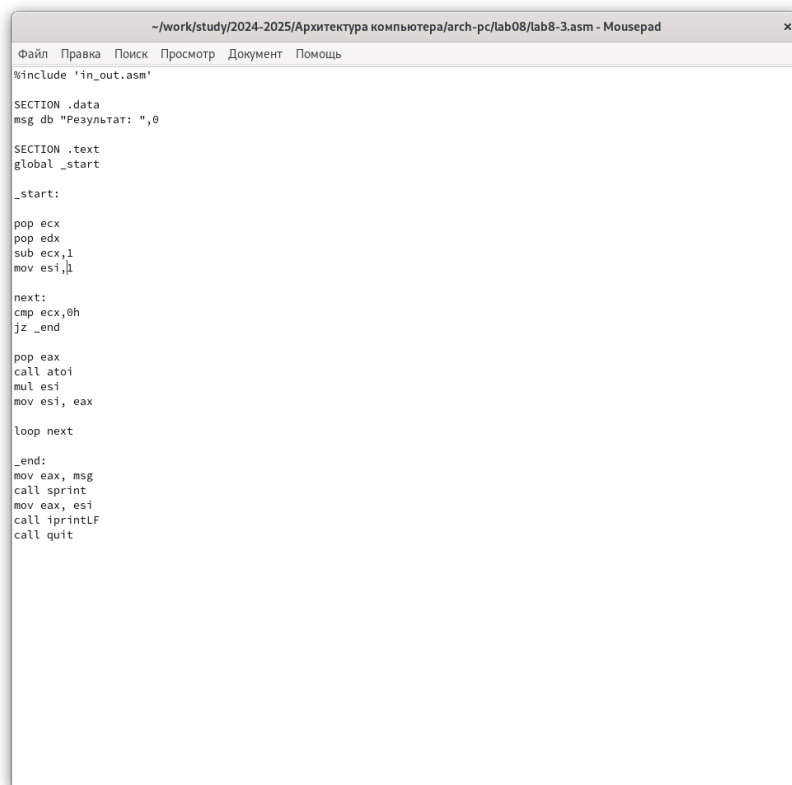
Компилирую программу и запускаю, указав в качестве аргументов некоторые числа, программа их складывает (рис. 4.11).



```
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура комп: klimenkoalena@fedora:~/work/study/2024-2025/Архитектура к
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ touch lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ mousepad lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_i386 -o lab8-3 lab8-3.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-3 12 13 7 10 5
Результат: 47
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$
```

Рис. 4.11: Запуск третьей программы

Изменяю поведение программы так, чтобы указанные аргументы она умножала, а не складывала (рис. 4.12).



```
~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08/lab8-3.asm - Mousepad
Файл  Правка  Поиск  Просмотр  Документ  Помощь
%include 'in_out.asm'

SECTION .data
msg db "Результат: ",0

SECTION .text
global _start

_start:
    pop ecx
    pop edx
    sub ecx,1
    mov esi,1

next:
    cmp ecx,0h
    jz _end

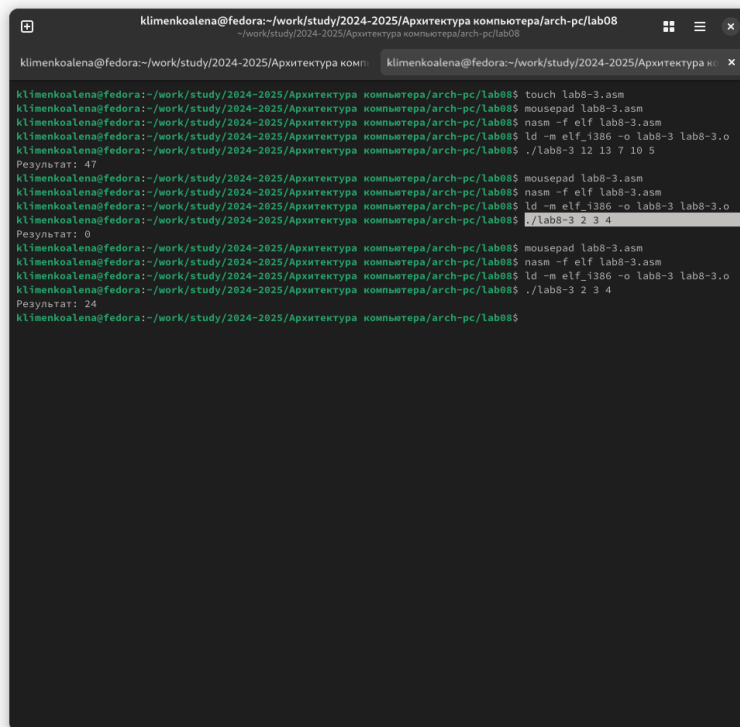
    pop eax
    call atoi
    mul esi, eax
    mov esi, eax

    loop next

_end:
    mov eax, msg
    call sprint
    mov eax, esi
    call iprintLF
    call quit
```

Рис. 4.12: Изменение третьей программы

Программа работает верно. (рис. 4.13).

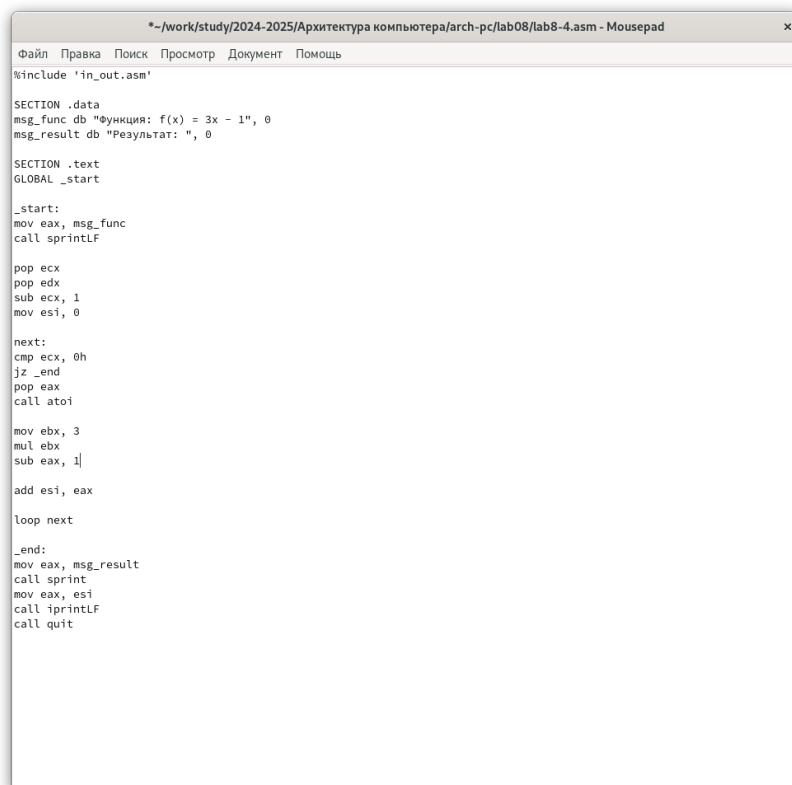


```
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура комп: klimenkoalena@fedora:~/work/study/2024-2025/Архитектура к
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ touch lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ mousepad lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ nasm -f elf lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ ld -m elf_i386 -o lab8-3 lab8-3.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ ./lab8-3 12 13 7 10 5
Результат: 47
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ mousepad lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ nasm -f elf lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ ld -m elf_i386 -o lab8-3 lab8-3.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ ./lab8-3 2 3 4
Результат: 0
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ mousepad lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ nasm -f elf lab8-3.asm
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ ld -m elf_i386 -o lab8-3 lab8-3.o
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$ ./lab8-3 2 3 4
Результат: 24
klimenkoalena@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/Lab08$
```

Рис. 4.13: Запуск измененной третьей программы

4.3 Задание для самостоятельной работы

Пишу программму, которая будет находить сумма значений для функции $f(x) = 3x-1$, которая совпадает с моим девытым варинтом (рис. 4.14).



```
*~\work\study\2024-2025\Архитектура компьютера\arch-pc\lab08\lab8-4.asm - Mousepad
Файл  Правка  Поиск  Просмотр  Документ  Помощь

%include 'in_out.asm'

SECTION .data
msg_func db "Функция: f(x) = 3x - 1", 0
msg_result db "Результат: ", 0

SECTION .text
GLOBAL _start

_start:
mov eax, msg_func
call sprintf

pop ecx
pop edx
sub ecx, 1
mov esi, 0

next:
cmp ecx, 0h
jz _end
pop eax
call atoi

mov ebx, 3
mul ebx
sub eax, 1

add esi, eax
loop next

_end:
mov eax, msg_result
call sprintf
mov eax, esi
call iprintf
call quit
```

Рис. 4.14: Написание программы для самостоятельной работы

Код программы:

```
%include 'in_out.asm'

SECTION .data
msg_func db "Функция: f(x) = 3x - 1", 0
msg_result db "Результат: ", 0

SECTION .text
GLOBAL _start

_start:
mov eax, msg_func
call sprintf
```

```

pop ecx
pop edx
sub ecx, 1
mov esi, 0

next:
cmp ecx, 0h
jz _end
pop eax
call atoi

mov ebx, 3
mul ebx
sub eax, 1

add esi, eax

loop next

_end:
mov eax, msg_result
call sprint
mov eax, esi
call iprintLF
call quit

```

Проверяю работу программы, указав в качестве аргумента несколько чисел (рис. 4.15).

```
klimenkoalea@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08
~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08
klimenkoalea@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ touch lab8-4.asm
klimenkoalea@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nousepad lab8-4.asm
klimenkoalea@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ nasm -f elf lab8-4.asm
klimenkoalea@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ld -m elf_i386 -o lab8-4 lab8-4.o
klimenkoalea@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$ ./lab8-4 1 2 3
Функция: f(x) = 3x - 1
Результат: 15
klimenkoalea@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc/lab08$
```

Рис. 4.15: Запуск программы для самостоятельной работы

5 Выводы

В результате выполнения данной лабораторной работы я приобрела навыки написания программ с использованием циклов а также научилась обрабатывать аргументы командной строки.

6 Список литературы

1. Курс на ТУИС
2. Лабораторная работа №8